

KEK Internal 99-4  
June 1999  
H/R/D

Lecture Note  
**Installation of EGS4**  
(Revised in 1999)

(Last modified on 04 OCT 2002)

Y. Namito, H. Nakamura and H. Hirayama

High Energy Accelerator Research Organization (KEK)

# Lecture Note

## Installation of EGS4

### (Revised in 1999)

Y. Namito, H. Nakamura and H. Hirayama  
*High Energy Accelerator Research Organization (KEK),  
Oho, Tsukuba-shi, Ibaraki-ken, 305, Japan*

#### Abstract

KEKでは、NRCCから配布している `egs4unix2.0` から必要なファイルを選び、KEKで開発した拡張機能を追加し、さらに多くのワークステーションで使用可能な C shell script を添付し、`egs4unix.kek` として配布している。また、PC用バッチファイルを添付し `egs4pc.kek` として、Macintosh用 shell script を添付し `egs4macintosh.kek` として配布している。本レポートでは、`egs4unix.kek`、`egs4pc.kek` および `egs4macintosh.kek` から unix ワークステーション (SUN、HP、DEC-ULTRIX、Silicon-Graphics、IBM、日立および Linux)、PC(DOS/V および 98) および Macintosh に EGS4 システムをインストールする手順をのべる。EGS4 システムは、MORTRAN、EGS4、PEGS4 およびユーザーコードよりなっている。EGS4 のインストールの目標は、`ucsamp14.mor` というユーザーコードを動かして SLAC-265 に示されている出力と同一の出力を得る事である。PEGS4 のインストールの目標は、本説明に示す `examin.mor` の出力と同等の出力を得る事である。PC では `g77`、Lahey Fortran LF90、Microsoft Fortran/Power station または Visual Fortran を、Macintosh では LS-Fortran と MPW shell をそれぞれ使用するとしてインストール方法を述べる。本レポートで対象としたコンパイラ以外のコンパイラを用いる場合に参考になる事も述べたので、本レポートはそのような場合にも役立つはずである。なお、本レポートでは同じ内容が日本語と英語の両方で述べられている。

# Contents

<b>Japanese Part</b>	<b>1</b>
<b>1 Unix 計算機へのインストール</b>	<b>1</b>
1.1 システム入手	1
1.2 ファイルの展開	1
1.3 MORTRAN のインストール	1
1.3.1 mortran3.dat の作成 (モード I、機種依存)	2
1.3.2 コンパイラオプションの変更 (機種依存)	3
1.3.3 check77.mor による mortran3(モード II) の動作確認	4
1.4 EGS4 のインストール	4
1.4.1 egs4run による実行	4
1.4.2 コマンドによる実行	5
1.5 PEGS4 のインストール	6
1.5.1 PEGS4 実行可能ファイルの作成	6
1.5.2 PEGS4 の実行	7
1.5.3 PEGS4 の動作確認	7
1.6 egs4_unix	8
<b>2 PC へのインストール</b>	<b>9</b>
2.1 システム入手	9
2.2 ファイルの展開	9
2.3 MORTRAN のインストール	9
2.3.1 MORTRAN3.DAT の作成 (モード I、機種依存)	9
2.3.2 コンパイラ名称およびオプションの変更 (機種依存)	11
2.3.3 check77.mor による mortran3(モード II) の動作確認	11
2.4 EGS4 のインストール	12
2.4.1 EGS4RUN による実行	12
2.4.2 コマンドによる実行	12
2.5 PEGS4 のインストール	13
2.5.1 PEGS4 実行可能ファイルの作成	13
2.5.2 PEGS4 の実行	14
2.5.3 PEGS4 の動作確認	14
<b>3 Macintosh へのインストール</b>	<b>15</b>
3.1 システム入手	15
3.2 ファイルの展開	15
3.3 MORTRAN のインストール	15
3.3.1 mortran3.dat の作成 (モード I、機種依存)	16
3.3.2 コンパイラオプションの変更 (機種依存)	17

3.3.3	check77.mor による mortran3 (モード II) の動作確認	17
3.4	EGS4 のインストール	18
3.4.1	egs4run による実行	18
3.4.2	コマンドによる実行	19
3.5	PEGS4 のインストール	20
3.5.1	PEGS4 実行可能ファイルの作成	20
3.5.2	PEGS4 の実行	21
3.5.3	PEGS4 の動作確認	21
<b>4</b>	<b>練習問題</b>	<b>22</b>
4.1	実習時の注意事項	22
4.2	EGS4RUN で生成するファイル	22
4.3	MORTRAN の練習問題	23
4.4	EGS4, PEGS4 の練習問題	23
4.5	MORTRAN の練習問題の解答	25
4.6	EGS4, PEGS4 の練習問題の解答	26

# Japanese Part

## 1 Unix 計算機へのインストール

### 1.1 システム入手

高エネ研のFTPサーバーから `egs4unix_kek` を `anonymous ftp` で入手する。ノード名: `ftp.kek.jp` (130.87.34.29) に `ftp` 接続し、ユーザーID を「`anonymous`」、パスワードとして「`自分のID@ノード名`」を入力してログインする。そして、

```
ftp> binary
```

でバイナリーモードに設定した後、次のファイルを `get` コマンドで入手する。

```
/kek/kek_egs4/egs4unix_kek/egs4unix_kek.tar.Z    (EGS4 システム)  
/kek/kek_egs4/egs4unix_kek/user_code.tar.Z      (EGS4 サンプルユーザーコード)
```

### 1.2 ファイルの展開

入手したファイルは `compress` コマンドで圧縮した `tar` ファイルである。作業は、ファイルを展開するディレクトリで実行する。

```
% uncompress egs4unix_kek.tar.Z  
% uncompress egs4user_code.tar.Z
```

とすると、ファイルの圧縮を解き、「`.Z`」の付かないファイルを生成するので、これらを

```
% tar xvf egs4unix_kek.tar  
% tar xvf user_code.tar
```

として個々のファイルに展開する。

以後、入力は「`%`」プロンプトで始まる行で示す。

### 1.3 MORTRANのインストール

MORTRANは `mortran3.f` と `Mortran` を Fortran に変換するための規則を含むデータファイル `mornew77.raw` よりなっている。`mortran3.f` をコンパイルして作成する実行形式ファイル `mortran3` は、次の二つのモードで動く。

**モード I:** `mortran3.dat` 作成モード

Mortran インストール時に使用する。`mornew77.raw` を16進形式に変換し `mortran3.dat` を作成する。

**モード II:** `mortran/fortran` 変換モード

EGS4 計算時に使用する。`mortran3.dat` を用いて `Mortran` で書かれたファイルを Fortran に変換する。

### 1.3.1 mortran3.dat の作成 (モード I、機種依存)

配布時に mortran3.f はモード I に設定されているので、そのままコンパイルする。

```
% f77 -o mortran3 -O mortran3.f          (SUN の場合)
% f77 -o mortran3 -OK mortran3.f        (HP の場合)
% f77 -o mortran3 -static -O mortran3.f  (DEC ULTRIX の場合)
% f77 -o mortran3 -static mortran3.f    (Silicon Graphics の場合)
% f77 -o mortran3 mortran3.f            (IBM AIX の場合)
% f77 -o mortran3 mortran3.f            (日立 HI-UX の場合)
% g77 -o mortran3 -O -fno-automatic mortran3.f (Linux の場合)
```

機種によってオプションが異なるので注意を要する。オプションの指定は2個の目的で行っている。

- 最適化。「O」(アルファベット)は最適化を指定している。日立のHI-UXの場合には、最適化が既定値で設定されている。Optimizeは、計算結果がDefaultの場合と有意に変わらない事を確認しながら行う。
- Local変数の保持。HPの場合には「-K」、DEC ULTRIXおよびSilicon Graphicsの場合には「-static」、Linuxのg77の場合には「-fno-automatic」によってLocal変数の保持を指示する必要がある。SUNと日立の場合にはDefaultでLocal変数が保持されるのでこれについての指定は必要ない。

上に列挙した以外のコンパイラを用いる場合にはこれらの機能を有効にするオプションを指定する。

コンパイル後、mortran3という実行ファイルができる。

```
% mortran3
```

と入力するとこのmortran3が実行され、mortran3.datが作成される。

作成されたmortran3.datの一行目、

```
v....2C USER F77 11JUN85
```

は削除する必要があるので行全体を削除する。

```
% vi mortran3.dat
```

でエディタを開き、

```
:1d
```

とキーボードからタイプすると1行目が削除される。(1行目に空白行を残さないよう注意する。)さらに、

```
:wq
```

とタイプするとエディタは同じファイル名に変更を保存した後終了し、Unixのプロンプトに戻る。なお、ここで、mortran3.datが作成されなかった場合には、viエディタでmortran3.fを開き、

```
IMODE=1
```

に設定されている事を確認する。また、CALL IOINITという文があれば、SUN以外では動かないのでコメントアウトする。

インストール時にmortran3.datを作成すれば、以後mortran3をモードIで使用することはない。通常のEGS4計算で使用するために、mortran3をモードIIに変更する。エディタでmortran3.fを開き、IMODE=1を

```
IMODE=2
```

と変更したのち前述の方法で再度コンパイルする。

```
% f77 -o mortran3 -O mortran3.f          (SUN の場合 )
% f77 -o mortran3 -OK mortran3.f         (HP の場合 )
% f77 -o mortran3 -static -O mortran3.f  (DEC ULTRIX の場合)
% f77 -o mortran3 -static mortran3.f     (Silicon Graphics の場合)
% f77 -o mortran3 mortran3.f            (IBM AIX の場合)
% f77 -o mortran3 mortran3.f            (日立 HI-UX の場合)
% g77 -o mortran3 -O -fno-automatic mortran3.f (Linux の場合)
```

新たに作成されたmortran3はモードIIで動作する。

### 1.3.2 コンパイラオプションの変更(機種依存)

日立HI-UXを用いての実習時にはスキップする。

日立HI-UXまたはIBM-AIX以外のWSを使用する場合には、mortrun、egs4runおよびpegs4compile中の

f77 mortjob.fという行を前節のコンパイルと同様に

```
f77 -O          mortjob.f  (SUN の場合)
f77 -OK        mortjob.f  (HP の場合)
f77 -static -O mortjob.f  (DEC ULTRIX の場合)
f77 -static    mortjob.f  (Silicon Graphics の場合)
g77 -O -fno-automatic mortjob.f (Linux の場合)
```

と変更する。この他に、日立やIBMのUnix機では、Auto-doubleのオプションも用意されているので必要に応じて利用できる。

上に列挙した以外のコンパイラを使用する場合には同等のオプションを捜して、それを指定する必要がある。前節のmortran3.fコンパイル時のオプション変更とこれらのファイル内のコンパイラオプションの変更が唯一の機種依存の部分であり、以後の操作はすべて、機種非依存である。

### 1.3.3 check77.mor による mortran3(モード II)の動作確認

check77.mor は Mortran の動作を確認するための長さ約 10 行のプログラムである。これが正常に Fortran プログラムに変換され、コンパイル・実行されれば、MORTRAN のセットアップが正常に行われたと判断できる。check77.mor の実行には、

```
% mortrun check77.mor
```

とする。mortrun は、Mortran 言語で書いたプログラムを Fortran に変換、コンパイル・実行するためのスクリプトである。(mortrun の 6 文字目が a ではなく u であることに注意) 実行が終わると、

```
**** Results of check77.f ***
THE FIRST TEN INTEGERS AND THEIR SQUARES:

I=  1 ( ODD INTEGER)    1
I=  2 (EVEN INTEGER)    4
I=  3 ( ODD INTEGER)    9
I=  4 (EVEN INTEGER)   16
I=  5 ( ODD INTEGER)   25
I=  6 (EVEN INTEGER)   36
I=  7 ( ODD INTEGER)   49
I=  8 (EVEN INTEGER)   64
I=  9 ( ODD INTEGER)   81
I= 10 (EVEN INTEGER)  100
```

とディスプレイ上に出力される。出力がこのようにならない場合には、mortjob.f と mortjob.list をエディタで見直し、マクロ等がすべて Fortran 文に変換されているか確認する。

## 1.4 EGS4のインストール

インストール時に、EGS4 本体のファイルを変更する必要はない。圧縮された tar file から展開して得られた egs4.mor 等が正常に動作するかのみを確認する。

### 1.4.1 egs4run による実行

egs4 の動作確認を目的として、egs4run という C shell script を用いサンプルユーザーコードである ucsampl4.mor を実行する。物質データはやはりサンプルである sampl4.dat を用いる。

egs4run を次のように対話式に実行する;

```
% egs4run ucsampl4.mor sampl4.dat
Do you want to include MACRO in mortran list? yes
Do you use PRESTA? no
```

これらを egs4run 起動時に指定しなければ、egs4run が対話式にそれらをたずねてくるので、それに答える形で動かしてもよい。egs4run が実行できない場合にはその実行権を調べる。図 1 に egs4run のリストを示す。

対話形式の部分が終了すると、

```
mortran3 procedure entered
f77 procedure entered
a.out procedure entered
egs4run procedure finished
```

と表示されプロンプトが表示される。ここで、

```
% ps
```

として、a.outの実行状況を確認する。実行中には、

```
PID    TT  STAT  TIME COMMAND
***   **   ***   *:* a.out
```

と、実行中のファイルとそれに要したCPU時間が打ち出される。

g77を使用する場合には、次のエラーメッセージが出力される。これは無視する。

```
mortjob.for: In block-data '_BLOCK_DATA_': mortjob.for:468: warning:
COMMON/ELECIN/ EKELIM, EKE0(10),EKE1(10),CMFPO(10),CMFP1(10),RANGE
Initialization of large (108772-unit) aggregate area 'elecín' at (^
currently very slow and takes lots of memory during g77 compile -- to be
improved in 0.6
```

実行が終了したら、mortjob.output<sup>1</sup>に出力される計算結果をエディタで見、図4に示すSLAC-265のucsampl4.morの出力と比較する。両方が同じであれば、EGS4が正しくインストールされた事になる。

今後、自分で作成したユーザーコード及びマテリアルデータ(PEGS4の項を参照)を使ってEGS4計算を行う場合も同じようにegs4runを用いる。

出力が図4と異なる場合や、出力が途中で止まっている場合またはエラーが起こった場合には、インストールが正しく行われていないので原因を調べる。この場合、他の出力ファイル(内容は第4.2節参照)の中を調べたり、次節で述べるようにegs4runの内容を一つ一つコマンドで実行したりして、起こった事を把握し原因を取り除く。

#### 1.4.2 コマンドによる実行

この節は、通常はスキップする。前節のegs4runでの実行がうまく行かなかった場合や、egs4runで行っている内容を知りたい場合のみこの節の作業を行う。

ユーザーコードucsampl4.morをUnixコマンドで走らせ、SLAC-265と同一の出力を得る手順を述べる。EGS4の実行は、ユーザーコードとEGS4本体との接続、Mortran FileのFortran Fileへの変換、Fortran Fileの実行という3部分に分割できる。まず、ucsampl4.morのStep.1に次のopen文を含むマクロがある事を確かめ、これらが無い場合には追加する。

```
open(unit=6 ,file='mortjob.output');
open(unit=8 ,file='mortjob.dummy');
open(unit=12,file='mortjob.xsec');
```

<sup>1</sup>この拡張子は、output6やout等としてもよい。これを変更するには、ucsampl4.morのunit 6のopen文を変更する。

また、STEP 2に\$OPEN; という行が有る事を確認する。(この行は、STEP 1に含まれるOPEN文で置き換えられるためのテンプレートである。)

次に、ユーザーコードとegs4本体を一つのファイルにまとめmortjob.mortranという名前のファイルとする。また、物質データファイルをmortjob.xsecという名前のファイルにコピーする。

```
% cat egs4mac.mortran kek4mac.mortran ucsampl4.mor\
% egs4blok.mortran egs4.mortran > mortjob.mortran
% cp sampl4.dat mortjob.xsec
```

ここで \ 記号はコマンド行の継続を示す。端末、プリンタによっては ¥で表示されることもある。

```
% mortran3
```

と入力すれば前節で作成した mortran3 が走り、mortjob.mortran を読み込みmortjob.fを作成するのでコンパイル・実行する；

```
% f77 mortjob.f
% a.out
```

これで、前節に示したものと同じ出力が mortjob.output に得られる。

## 1.5 PEGS4のインストール

### 1.5.1 PEGS4 実行可能ファイルの作成

PEGS4は通常変更しないので、mortran3と同様に実行可能ファイルの形で保存する。コンパイルにはpegs4compileを用いる(内容は図5参照)

```
% pegs4compile
```

と入力するとpegs4.mortranをコンパイルし、作成された実行可能ファイルをpegs4.exeとして保存する。

コンパイルに失敗した場合には、pegs4.mortranをエディタで開きCALL IOINITがコメントアウトしてある事を確認する事。その時に、次のOPEN文が書いてあることも確認する事。

```
OPEN(5, file='pgs4job.pegs4inp');
OPEN(6, file='pgs4job.pegs4lst');
OPEN(7, file='pgs4job.pegs4dat');
OPEN(8, file='pgs4pepr.dat');
OPEN(9, file='pgs4form.dat');
OPEN(10, file='pgs4job.pegs4err');
OPEN(21, file='pgs4job.pegs4plot');
OPEN(22, file='aprime.data');
```

### 1.5.2 PEGS4の実行

pegs4run を用いて peps4.exe を実行し、物質データを作成する。通常は PEGS4 の入力データを用意するが、ここでは、saml4.inp (内容は図 9 参照) を入力データとして試しに実行する。

```
% peps4run saml4.inp peps4.output
```

ここで、第 1 オペラントが入力ファイル名であり第 2 オペラントが出力ファイル名である。pegs4run が実行できない場合には、その実行権を

```
% ls -l peps4run
```

で調べる。

### 1.5.3 PEGS4の動作確認

PEGS4 は自動的にエネルギー分点を設定し、その分点の間での物理量 (全断面積・分岐比など) を直線内挿しその直線の係数を出力する。インストールを行う計算機上での PEGS4 の動作を調べるために基準となる計算機での PEGS4 出力とインストールを行う計算機での PEGS4 出力を較べることが必要であるが、出力を直接較べるのは困難である。計算精度の違いなどによりエネルギー分点が違っていると同一物理量でも見かけ上全く違う表現になるからである。このため、examin.mor という EGS4 のユーザーコードを使用する。examin.mor は、PEGS4 データを読み込みそれを表の形で出力し SHOWER を CALL せずに終了するユーザーコードで、

```
% egs4run examin.mor peps4.output
```

と入力すれば実行できる。(PRESTA は使用しない)

ここで作成された mortjob.output6 と examin.out を

```
% diff examin.out mortjob.output6
```

で比較し、全く同一であるか、5桁目が何箇所か違う位であれば、インストールをおこなった計算機上で PEGS4 が、標準となる WS 上での PEGS4 と同じ動作をしたと考えられる。明かに異なれば原因をさがす必要がある。examin.out は、saml4.inp を入力とし Storm and Israel が編集した光電効果断面積 (pgs4pepr.dat) を用いて、PEGS4 を KEK の共通計算機 (ccce5=日立製 WS) で走らせて得たファイルの examin.mortran による分析結果であり、その内容を章末 (図 10 および 11) に示す<sup>2</sup>。

---

<sup>2</sup>Storm & Israel の編集値に基づく光電効果及び電子対生成断面積データ (pgs4pepr.dat) を PEGS4 では使用してきており、egs4unxi.kek、egs4pc.kek および egs4macintosh.kek でもデフォルトで、この pgs4pepr.dat を使用している。一方、原研・坂本氏は、最近の光子断面積データである PHOTX を用い新しい光電効果及び電子対生成断面積データファイル pgs4phtx.dat を作成した (「第 3 回 EGS4 研究会」, KEK Proceedings 93-15 (1993) 77-82 参照)。egs4unxi.kek、egs4pc.kek および egs4macintosh.kek には pgs4phtx も含まれている。pgs4pepr.dat の代わりに pgs4phtx.dat を使用するには、pegs4.mor 内の open 文 (unit=8) のファイル名を pgs4pepr.dat から pgs4phtx.dat に変更する。

## 1.6 egs4\_unix

カナダのNRCCとLBLが、egs4\_unix ver.3.0を配布している。これは、SLACで開発されたEGS4に低エネルギーの電子の輸送に関する改良(PRESTA等)を加え、さらにUNIX(SUN、IRIS、HP、DEC、IBM Rシリーズ、Linuxなど)で動くように修正を加えたものである。

egs4\_unix ver.3.0は対象としている計算機へのインストールが自動的に行え便利である反面、ユーザーコード内で外部ファイルを参照する際にOPEN文の使用を避けるために各機種固有のコマンドを用いているため、対象とする計算機以外の計算機では、C shell scriptの変更が必要になる。また、コンパイラやOSが変更されたときにも、C shell scriptの変更が必要となる。ところが、添付されているC shell scriptがいずれも数100行と大きく、変更が容易ではない。これに対してKEK版の機種依存はコンパイラオプションの指定のみである。従って、NRCC版とKEK版を比較すると、NRCC/LBLの配布しているものは、問題がおきなければ全自動でインストールが行え便利であるが、問題が発生したときに対処に手間取る面がある。一方、KEK版は常にインストールに一定の作業を要するが、OSやコンパイラの変更が有っても容易に対処できるという利点がある。

egs4unix ver.3.0には、MORTRAN、PEGS4の実行可能ファイル、egs4のコンパイル及び実行用egs4.compile及びegs4.runも含まれている。NRCC版egs4unix ver3.0が必要な場合には、EGS4ホームページ(<http://ehssun.lbl.gov/egs/egs.html>)にアクセスする。

## 2 PCへのインストール

### 2.1 システム入手

高エネ研のFTPサーバーから `egs4pc_kek` を `anonymous ftp` で入手する。ノード名: `ftp.kek.jp` (130.87.34.29) に `ftp` 接続し、ユーザーIDを「`anonymous`」、パスワードとして「自分のID@ノード名」を入力してログインする。そして、

```
ftp> binary
```

でバイナリーモードに設定した後、次のファイルを `get` コマンドで入手する。

```
/kek/kek_egs4/egs4pc_kek/egs4pc.exe (EGS4 システム)  
/kek/kek_egs4/egs4pc_kek/user_cod.exe (EGS4 サンプルユーザーコード)
```

### 2.2 ファイルの展開

入手したファイルはLHAによる自己解凍形式ファイルである。これらのファイルを作業を行うディレクトリ(例えば、`C:\%EGS4LECT`)に移動し、

```
C:\%EGS4LECT> EGS4PC.EXE  
C:\%EGS4LECT> USER_COD.EXE
```

として個々のファイルに展開する。

以後、入力は「`C:\%EGS4LECT`」プロンプトの行で示す。

### 2.3 MORTRANのインストール

MORTRANはMORTRAN3.FORとMortranをFortranに変換するための規則を含むデータファイルMORNEW77.RAWよりなっている。MORTRAN3.FORをコンパイルして作成する実行形式ファイルMORTRAN3.EXEは、次の二つのモードで動く。

モード I: MORTRAN3.DAT 作成モード

Mortran インストール時に使用する。MORNEW77.RAWを16進形式に変換しMORTRAN3.DATを作成する。

モード II: mortran/fortran 変換モード

EGS4計算時に使用する。MORTRAN3.DATを用いてMortranで書かれたファイルをFortranに変換する。

#### 2.3.1 MORTRAN3.DATの作成(モード I、機種依存)

配布時にMORTARN3.FORはモード Iに設定されているので、そのままコンパイルする。<sup>3</sup>

---

<sup>3</sup>この前に、FORTRANコンパイラをインストールし、使用可能な状態にしておく。たとえば、Windows用g77 (<http://www.geocities.com/Athens/Olympus/5564>) を `C:\%g77` にインストールした場合には、DOS窓で、

```
C:\%EGS4LECT> c:\%g77\%g77setup
```

として初期化を行う。詳しくは各コンパイラのマニュアルを参照。

```

C:¥EGS4LECT> LF90 MORTRAN3.FOR (Lahey Fortran の場合)
C:¥EGS4LECT> FL32 MORTRAN3.FOR (MS-Fortran/PS の場合)
C:¥EGS4LECT> F90 MORTRAN3.FOR (Visual Fortran Ver.5 の場合)
C:¥EGS4LECT> DF MORTRAN3.FOR (Visual Fortran Ver.6.1 の場合)
C:¥EGS4LECT> g77 -o mortran3 -O -fno-automatic -fno-backslash mortran3.for
(g77 の場合)

```

コンパイル後、MORTARAN3.EXE という実行ファイルができるので、これを

```
C:¥EGS4LECT> MORTRAN3
```

と入力して実行すると、MORTRAN3.DAT が作成される。作成された MORTRAN3.DAT の一行目、

```
v....2C USER F77 11JUN85
```

は削除する必要があるがあるので MIFES または WZ 等のエディタで行全体を削除する。(1 行目に空白行を残さないよう注意する。)

なお、ここで、MORTRAN3.DAT が作成されなかった場合には、エディタで MORTRAN3.FOR を開き、

```
IMODE=1
```

に設定されている事を確認する。また、CALL IOINIT という文があれば、SUN 以外では動かないのでコメントアウトする。

インストール時に MORTRAN3.DAT を作成すれば、以後 MORTRAN3 をモード I で使用することはない。通常の EGS4 計算で使用するために、MORTRAN3 をモード II に変更する。エディタで MORTRAN3.FOR を開き、IMODE=1 を

```
IMODE=2
```

と変更したのち前述の方法で再度コンパイルする。

```

C:¥EGS4LECT> LF90 MORTRAN3.FOR (Lahey Fortran の場合)
C:¥EGS4LECT> FL32 MORTRAN3.FOR (MS-Fortran/PS の場合)
C:¥EGS4LECT> F90 MORTRAN3.FOR (Visual Fortran Ver.5 の場合)
C:¥EGS4LECT> DF MORTRAN3.FOR (Visual Fortran Ver.6.1 の場合)
C:¥EGS4LECT> g77 -o mortran3 -O -fno-automatic -fno-backslash mortran3.for
(g77 の場合)

```

新たに作成された MORTRAN3.EXE はモード II で動作する。

### 2.3.2 コンパイラ名称およびオプションの変更 (機種依存)

g77を使用する場合は、この節はスキップする。

次節で用いる MORTRUN.BAT は g77 用の設定になっているので、それ以外のコンパイラを使用する場合には、MORTRUN.BAT 内の g77 コンパイラの行をコメントアウトし、使用するコンパイラの行を使用可能にする事。後で使用する EGS4RUN.BAT、EGS4RUNP.BAT および PEGS4COM.BAT についてもここで同様の変更を行う事。

バッチファイルにかいてあるコンパイラ以外のコンパイラを使用する場合には、これらの Batch ファイル中のコンパイラ名およびオプションを変更する事。Optimize のレベルは計算結果が Default の場合と有意に変わらない事を確認しながら段階的にあげていく事。PEGS4COM.BAT は実行時間がかからないので最適化の必要はない。Local 変数の保持が default になっていない場合にはそれを MORTRUN.BAT と EGS4RUN.BAT および EGS4RUNP.BAT 内の Fortran コンパイルのオプションとして指定する必要がある (第 1.3.2 節参照)

PCを使用する場合の機種依存はこのコンパイラ名およびコンパイラオプションと第 2.5 で述べる namelist 入力の区切り記号の違いの 2点のみである。

### 2.3.3 check77.mor による mortran3(モード II)の動作確認

CHECK77.MOR は Mortran の動作を確認するための長さ約 10 行のプログラムである。これが正常に Fortran プログラムに変換され、コンパイル・実行されれば、MORTRAN のセットアップが正常に行われたと判断できる。CHECK77.MOR の実行には、

```
C:¥EGS4LECT> MORTRUN CHECK77.MOR
```

とする。MORTRUN.BAT は、Mortran 言語で書いたプログラムを Fortran に変換、コンパイル・実行するためのスクリプトである。(MORTRUN の 6 文字目が A ではなく U であることに注意)

実行が終わると、

```
**** Results of check77.f ***
THE FIRST TEN INTEGERS AND THEIR SQUARES:

I=  1 ( ODD INTEGER)      1
I=  2 (EVEN INTEGER)      4
I=  3 ( ODD INTEGER)      9
I=  4 (EVEN INTEGER)     16
I=  5 ( ODD INTEGER)     25
I=  6 (EVEN INTEGER)     36
I=  7 ( ODD INTEGER)     49
I=  8 (EVEN INTEGER)     64
I=  9 ( ODD INTEGER)     81
I= 10 (EVEN INTEGER)    100
```

とディスプレイ上に出力される。出力がこのようなにならない場合には、MORTJOB.FOR と MORTJOB.MLS をエディタで見直し、マクロ等がすべて Fortran 文に変換されているか確認する。

## 2.4 EGS4のインストール

インストール時に、EGS4本体のファイルを変更する必要はなく、展開して得られたEGS4.MOR等が正常に動作するかのみを確認する。

### 2.4.1 EGS4RUNによる実行

egs4の動作確認を目的として、EGS4RUN.BATというバッチファイル(内容は図2参照)を用いサンプルユーザーコードであるUCSAMPL4.MORを実行する。物質データはやはりサンプルであるSAMPL4.DATを用いる。

```
C:¥EGS4LECT> EGS4RUN UCSAMPL4.MOR SAMPL4.DAT
```

を実行し、実行が終了するまで待つ。

g77を使用する場合には、次のエラーメッセージが出力される。これは無視する。

```
mortjob.for: In block-data '_BLOCK_DATA_': mortjob.for:468: warning:
COMMON/ELECIN/ EKELIM, EKE0(10),EKE1(10),CMFPO(10),CMFP1(10),RANGE
Initialization of large (108772-unit) aggregate area 'elecjin' at (^)
currently very slow and takes lots of memory during g77 compile -- to be
improved in 0.6
```

実行が終了したら、MORTJOB.OUT<sup>4</sup>に出力される計算結果をエディタで見、図4に示すSLAC-265のucsampl4.morの出力と比較する。両方が同じであれば、EGS4が正しくインストールされた事になる。

自分で作成したユーザーコード及びマテリアルデータ(PEGS4の項を参照)を使ってEGS4計算を行う場合も同じようにEGS4RUN.BATをもちいる。なお、PCでPRESTAを使用する場合にはEGS4RUNPを用いる<sup>5</sup>。

出力が図4と異なる場合や、出力が途中で止まっている場合またはエラーが起こった場合には、インストールが正しく行われていないので原因を調べる。この場合、他の出力ファイル(内容は第4.2節参照)の中を調べたり、次節で述べるようにEGS4RUN.BATの内容を一つ一つコマンドで実行したりして、起こった事を把握し原因を取り除く。

### 2.4.2 コマンドによる実行

この節は、通常はスキップする。前節のEGS4RUN.BATでの実行がうまく行かなかった場合や、EGS4RUN.BATで行っている内容を知りたい場合のみこの節の作業を行う。

ユーザーコードUCSAMPL4.MORをDOSコマンドで走らせ、SLAC-265と同一の出力を得る手順を述べる。EGS4の実行は、ユーザーコードとEGS4本体との接続、Mortran FileのFortran Fileへの変換、Fortran Fileの実行という3部分に分割できる。まず、ucsampl4.morのStep.1に次のopen文を含むマクロがある事を確かめ、これらが無い場合には追加する。

<sup>4</sup>拡張子は、ou6やout等としてもよい。これを変更するには、ucsampl4.morのunit 6のopen文を変更する。

<sup>5</sup>例えば、C:¥EGS4LECT> EGS4RUNP UCNAI.MOR NAI.DATとする。

```
open(unit=6 ,file='mortjob.out');
open(unit=8 ,file='mortjob.dum');
open(unit=12,file='mortjob.xse');
```

また、STEP 2に\$OPEN; という行が有る事を確認する。(この行は、STEP 1に含まれるOPEN文で置き換えられるためのテンプレートである。)

次に、ユーザーコードとegs4本体を一つのファイルにまとめMORTJOB.MORという名前のファイルとする。また、物質データファイルをMORTJOB.XSEという名前のファイルにコピーする。

```
C:¥EGS4LECT>COPY LISTING.ON+EGS4MAC.MOR+NRCC4MAC.MOR+PAIRMAC.MOR+
KEK4MAC.MOR+UCSAMPL4.MOR+KEK4.MOR+EGS4BLOK.MOR+EGS4.MOR MORTJOB.MOR
C:¥EGS4LECT>COPY SAMPL4.DAT MORTJOB.XSE
```

```
C:¥EGS4LECT>MORTRAN3
```

と入力すれば前節で作成したMORTRAN3.EXEが走り、MORTJOB.MORを読み込みMORTJOB.FORを作成するのでコンパイル・実行する；

```
C:¥EGS4LECT>LF90 MORTJOB.FOR -nw -lst      (Lahey Fortranの場合)
C:¥EGS4LECT>FL32 /Ox /W0 /G4 MORTJOB.FOR  (MS-Fortran/PSの場合)
C:¥EGS4LECT>MORTJOB.EXE
```

これで、前節に示したものと同一出力がMORTJOB.OUTに得られる。

## 2.5 PEGS4のインストール

### 2.5.1 PEGS4 実行可能ファイルの作成

PEGS4は通常変更しないので、MORTRAN3.EXEと同様に実行可能ファイルの形で保存する。コンパイルにはPEGS4COM.BATを用いる(内容は図5参照)

```
C:¥EGS4LECT> PEGS4COM
```

と入力するとPEGS4.MORをコンパイルし、作成された実行可能ファイルをPEGS4.EXEとして保存する。

コンパイルに失敗した場合には、PEGS4.MORをエディタで開きCALL IOINITがコメントアウトしてある事を確認する事。その時に次のOPEN文が書いてあることも確認する事。

```
OPEN(5, file='pgs4job.inp');
OPEN(6, file='pgs4job.lst');
OPEN(7, file='pgs4job.dat');
OPEN(8, file='pgs4pepr.dat');
OPEN(9, file='pgs4form.dat');
OPEN(10, file='pgs4job.err');
OPEN(21, file='pgs4job.plt');
OPEN(22, file='aprime.dat');
```

### 2.5.2 PEGS4の実行

PEGS4RUN.BATを用いてPEGS4.EXEを実行し、物質データを作成する。通常はPEGS4の入力データを用意するが、ここでは、SAMPL4.INPを入力データとして試しに実行する。

```
C:¥EGS4LECT> PEGS4RUN SAMPL4.INP PEGS4.OUT
```

ここで、第1オペラントが入力ファイル名であり第2オペラントが出力ファイル名である。SAMPL4.INPの内容は図9に示す。MS-Fortran/Power Stationを使用する場合には、図9の後半に示すようにSAMPL4.INP内の「&END」を「/」に変更する必要がある。<sup>6</sup>

### 2.5.3 PEGS4の動作確認

PEGS4は自動的にエネルギー分点を設定し、その分点の間での物理量(全断面積・分岐比など)を直線内挿しその直線の係数を出力する。インストールを行う計算機上でのPEGS4の動作を調べるために基準となる計算機でのPEGS4出力とインストールを行う計算機でのPEGS4出力を較べる必要があるが、出力を直接較べるのは困難である。計算精度の違いなどによりエネルギー分点が違っていると同一物理量でも見かけ上全く違う表現になるからである。このため、EXAMIN.MORというEGS4のユーザーコードを使用する。EXAMIN.MORは、PEGS4データを読み込みそれを表の形で出力しSHOWERをCALLせずに終了するユーザーコードで、

```
C:¥EGS4LECT> EGS4RUN EXAMIN.MOR PEGS4.OUT
```

と入力すれば実行できる。(PRESTAは使用しない)

ここで作成された、MORTJOB.OUTとEXAMIN.OUTを

```
C:¥EGS4LECT>FC EXAMIN.OUT MORTJOB.OUT
```

で比較し、全く同一であるか、5桁目が何箇所か違う位であれば、インストールをおこなった計算機上でPEGS4が、標準となる計算機上でのPEGS4と同じ動作をしたと考えられる。明かに異なれば原因をさがす必要がある。EXAMIN.OUTは、SAMPL4.INPを入力としStom and Israelが編集した光電効果断面積(PGS4PEPR.DAT<sup>7</sup>)を用いて、PEGS4をKEKの共通計算機(ccce5=日立製WS)で走らせて得たファイルのEXAMIN.MORによる分析結果であり、その内容を章末(図10および11)に示す。

<sup>6</sup>MS-Fortran/Power Stationを使用する場合には、他のnamelist入力についてもすべてこの変更をする必要がある。

<sup>7</sup>PGS4PEPR.DATについては、「Unix計算機へのインストール」の部分の脚注2を参照。

### 3 Macintoshへのインストール

Macintosh用のLS FORTRANとMPW (Macintosh Programmer's Workshop) shellを使用する。MPW shellはLS Fortranの付属している。

#### 3.1 システム入手

高エネ研のFTPサーバーからMacintosh用EGS4システムをanonymous ftpで入手する。まず、Macintosh側で「FTP enable」と「MacBinary enable」を有効にする<sup>8</sup>。そしてノード名:ftp.kek.jp(130.87.34.29)にftp接続し、ユーザーIDを「anonymous」、パスワードとして「自分のID@ノード名」を入力する<sup>9</sup>。そして、

```
ftp> binary
```

でバイナリモードに設定した後、次のファイルをgetコマンドで入手する。

```
/kek/kek_egs4/egs4macintosh_kek/egs4macintosh.sea (EGS4システム)
/kek/kek_egs4/egs4macintosh_kek/user_code.sea (EGS4サンプルユーザーコード)
```

#### 3.2 ファイルの展開

入手したファイルはDrop Stuffによる自己解凍形式ファイルである。egs4macintosh.seaをMacintoshのハードディスクにコピーして、ファイルのアイコンをダブルクリックすると自動展開する。展開の時には最初にegs4macintosh folderが作成され、そのフォルダー内にファイルが展開される。(この時にフォルダー名を変えることもできる。)user\_code.seaも同様に展開する。

#### 3.3 MORTRANのインストール

LS FORTRANはMPWのshell上で動作をする。まず最初にMPWのアイコンをクリックしMPWのWorksheetを立ち上げる。以後の操作はMPW上で行う。

<sup>8</sup>これらは、入手するファイルが「自己解凍可能」という属性をもっている事を認識するために必要である。この属性が認識できないとファイルの展開ができない。なお、ftpはNCSA telnet Ver. 2.5を使用してテストした。

- Fetchを用いてもftpによるファイル入手とファイル展開が可能である。この場合はFetch内で「自己解凍可能」という属性を指定する。
- PCやunixを経由してファイルを入手した場合も、この「自己解凍可能」属性は消えてしまう。この場合はStuffit Expander Ver. 3.5.2を使用すればファイルを展開する事ができる(Ver. 4.0.2では展開ができない)。

<sup>9</sup>先にtelnetが動作しているとftpでの接続でlsコマンドが動作しなかった。このためtelnetを使用していない状態でftpを起動する必要がある。

MORTRANはmortran3.fとMortranプログラムをFortranプログラムに変換するための規則を含むデータファイルmornew77.rawよりなっている。mortran3.fをコンパイルして作成する実行形式ファイルmortran3は、次の二つのモードで動く。

#### モード I: mortran3.dat 作成モード

Mortran インストール時に使用する。mornew77.rawを16進形式に変換しmortran3.datを作成する。

#### モード II: mortran/fortran 変換モード

EGS4 計算時に使用する。mortran3.datを用いてMortranで書かれたファイルをFortranに変換する。

### 3.3.1 mortran3.dat の作成 (モード I、機種依存)

配布時にmortran3.fはモード Iに設定されているので、そのままコンパイルする。MPWでのコンパイルの仕方は次のとおり。DirectoryメニューのSet Directoryを選択し、mortran3.fの入っているdirectoryを設定する。Worksheet上で

```
RunPPC mortran3.f -saveall (enter)
```

と打ち込むと、mortran3.fのコンパイル、mortran3の実行が行われ、mortran3.datが作成される。ここで、コンパイラオプションの「-saveall」はローカル変数の保持を指示するために必要である。コマンドをタイプした後でReturnではなくenterキーを押す事およびファイル名の一番最後は拡張子「.f」を省略せずに付ける事に注意する。

作成されたmortran3.datの1行目、

```
v....2C USER F77 11JUN85
```

は削除する必要があるのでこの行全体を削除する。(1行目に空行を残さないこと。)

なお、ここでmortran3.datが作成されなかった場合には、mortran3.fを開き、

```
IMODE=1
```

に設定されている事を確認する。また、CALL IOINITという文があれば、SUN以外では動かないのでコメントアウトする。

インストール時にmortran3.datを作成すれば、以後mortran3をモード Iで使用することはない。通常のEGS4計算で使用するために、mortran3をモード IIに変更する。まず、mortran3.fを開き、IMODE=1を

```
IMODE=2
```

と変更して再度以下のようにコンパイルする。

```
RunPPC mortran3.f -saveall (enter)
```

新たに作成されたmortran3はモード IIで動作し、以後これをMortran → Fortran 変換用プログラムとして使う。

### 3.3.2 コンパイラオプションの変更(機種依存)

LS Fortran を使用する場合は、この節はスキップする。

LS Fortran 以外の Fortran コンパイラを用いる場合は、第 1.3.2 節と第 2.3.2 節に述べたコンパイラ名とコンパイラオプションの変更方法を参照する事。現時点では Macintosh 上で LS Fortran 以外の Fortran コンパイラをもちいては egs4 を動作させていないので unix と PC の所で述べた点以外にどのような機種、コンパイラ依存があるかは不明である。

### 3.3.3 check77.mor による mortran3 (モード II) の動作確認

次に mortran3 の動作を確認する。check77.mor は Mortran の動作を確認するための長さ約 10 行のプログラムである。これが正常に Fortran プログラムに変換され、コンパイル・実行されれば、MORTRAN のセットアップが正常に行われたと判断できる。check77.mor の実行には mortrun を利用する。

```
mortrun check77.mor
```

と打ち込んで実行する。(最後に enter キーを押すのを忘れずに！)

mortrun は Mortran 言語で書かれたプログラムを Fortran に変換、コンパイル、実行するための MPW-shell スクリプトである。(mortrun の 6 文字目が a ではなく u であることに注意) mortrun ファイルの内容は以下の通りである。

```
echo "mortrun has started"
delete mortjob.≈|| echo "There aren't any mortjob.≈"
catenate {1} >mortjob.mortran
mortran3
alert 'Wait until mortran3 finishes'
RunPPC mortjob.f -saveall
```

「≈」のマークはワイルドカードの意味を持つ。「alert 'Wait until mortran3 finishes'」の行は mortran3 が終了しないうちに「RunPPC mortjob.f -saveall」に進行するのを防ぐためのものである。「mortjob.≈」というファイルがひとつも存在しない時に、「delete mortjob.≈」を実行すると、コンピューターが異常動作を起こして止る。(ハングする。)これを防ぐため、delete 命令が失敗した場合に「||」の後の命令を実行するようにしてある。

実行が始まると、最初にビーブ音と共に

```
Wait until mortran3 finishes
```

と表示される。mortran3 が動作している間はディスプレイの右上に菱形状のアイコンが表示される。mortran3 の実行が終わると

```
Program executed STOP statement
```

と表示される。この OK をクリックし、次に

```
Wait until mortran3 finishes
```

のOKをクリックする。これにより、「RunPPC mortjob.f -saveall」が実行され、新たなWINDOWに

```
**** Results of check77.f ***
THE FIRST TEN INTEGERS AND THEIR SQUARES:

I= 1 ( ODD INTEGER)    1
I= 2 (EVEN INTEGER)    4
I= 3 ( ODD INTEGER)    9
I= 4 (EVEN INTEGER)   16
I= 5 ( ODD INTEGER)   25
I= 6 (EVEN INTEGER)   36
I= 7 ( ODD INTEGER)   49
I= 8 (EVEN INTEGER)   64
I= 9 ( ODD INTEGER)   81
I= 10 (EVEN INTEGER)  100
```

STOP

と表示される。ディスプレイの右上にはmortjob.exeがアクティブである事を示す菱形状のアイコンが表示される。この出力が得られればMORTRANのインストールが正常に終了したことになる。出力がこのようにならない場合には、mortjob.fとmortjob.listをエディタで見直し、マクロ等がすべてFortran文に変換されているか確認する。

なお間違えてmortrunと打ち込んだだけでenterキーを押すとWorksheetの左上に「catenate」と表示された状態で動作が止ってしまう。これはファイルをコピーするcatenateコマンドがファイルを特定できないためである。この状態から抜けるには（アップルキー：#の4隅を丸くした記号。または、テニスのラケットを4本組み合わせたような記号。通常スペースキーの両側にある。）と「.」（ピリオドキー）を同時に押す。すると、Worksheetに

```
### MPW Shell - Execution of mortrun terminated.
```

と表示して終了する。

### 3.4 EGS4のインストール

インストール時にEGS4本体のファイルを変更する必要はなく、展開して得られたegs4.mor等が正常に動作するかのみを確認する。

#### 3.4.1 egs4runによる実行

egs4の動作確認を目的として、egs4runというMPW-Shellスクリプトファイル（図3参照）を用いてサンプルユーザーコードであるucsampl4.morを実行し、結果が正しい事を確認する。物質データはやはりサンプルであるsampl4.datを用いる。

```
egs4run ucsampl4.mor sampl4.dat (enter)
```

を実行する。ファイル名を指定しないとmortrunの時と同様にcatenateのところで止ってしまう。その時には（アップルキー：#の4隅を丸くした記号。または、テニスのラケッ

トを4本組み合わせたような記号。通常スペースキーの両側にある。)と「.(ピリオドキー)を同時に押す。

実行が始まると、最初にピープ音と共に

```
Wait until mortran3 finishes
```

と表示される。mortran3が動作している間はディスプレイの右上に菱形状のアイコンが表示される。mortran3の実行が終わると

```
Program executed STOP statement
```

と表示される。このOKをクリックし、次に

```
Wait until mortran3 finishes
```

のOKをクリックする。これにより、「RunPPC mortjob.f -saveall」が実行される。mortjobの実行が終わるまで待つ。

計算結果はmortjob.outputに出力されるので、それを図4に示すSLAC-265のucsamp14.morの出力と比較する。両方が同じであれば、EGS4が正しくインストールされた事になる。

今後、自分で作成したユーザーコード及びマテリアルデータ(PEGS4の項を参照)を使ってEGS4計算を行う場合も同じようにegs4runを用いる。なお、PRESTAを使用するときにはegs4runのかわりにegs4runpを使用する<sup>10</sup>。

出力が図4と異なる場合や、出力が途中で止まっている場合またはエラーが起こった場合には、インストールが正しく行われていないので原因を調べる。この場合、他の出力ファイル(内容は第4.2節参照)の中を調べたり、次節で述べるようにegs4runの内容を一つ一つコマンドで実行したりして、起こった事を把握し原因を取り除く。

### 3.4.2 コマンドによる実行

この節は通常はスキップする。前節のegs4runでの実行がうまく行かなかった場合や、egs4runで行っている内容を知りたい場合のみこの節の作業を行う。

ユーザーコードucsamp14.morをMPWのSHELLコマンドで走らせ、SLAC-265と同一の出力を得る手順を述べる。EGS4の実行は、ユーザーコードとEGS4本体との接続、Mortran FileのFortran Fileへの変換、Fortran Fileの実行という3部分に分割できる。まず、ucsamp14.morのStep.1に次のopen文を含むマクロがある事を確かめ、これらが無い場合には追加する。

```
OPEN(6,FILE='mortjob.output',status='new');  
OPEN(8,FILE='mortjob.dummy',status='new');  
OPEN(12,FILE='mortjob.xsec',status='old');
```

また、Step.2に

<sup>10</sup>例えば、egs4runp ucna1.mor nai.dat とする。

```
$OPEN;
```

という行があることを確認する。(この行は、STEP 1に含まれる OPEN 文で置き換えられるためのテンプレートである。)

次にユーザーコードと egs4 本体を一つのファイルにまとめ mortjob.mortran という名前のファイルとする。また、物質データファイルを mortjob.xsec という名前のファイルにコピーする。各コマンドの最後に必ず enter キーを押す。

```
catenate listing.on > mortjob.mortran
catenate egs4mac.mortran >>mortjob.mortran
catenate nrcc4mac.mortran >>mortjob.mortran
catenate kek4mac.mortran >>mortjob.mortran
catenate ucsampl4.mor >>mortjob.mortran
catenate kek4.mortran >> mortjob.mortran
catenate egs4blok.mortran >> mortjob.mortran
catenate egs4.mortran >>mortjob.mortran
```

```
catenate sampl4.dat >mortjob.xsec
```

つぎに

```
mortran3.exe
```

と入力してこれを実行すると mortjob.mortran を読み込み mortjob.f を作成する。つぎに

```
RunPPC mortjob.f -saveall -opt=3
```

と入力して mortjob.f をコンパイル、実行する。

これで、前節に示したものと同一出力が mortjob.output に得られる。

### 3.5 PEGS4 のインストール

#### 3.5.1 PEGS4 実行可能ファイルの作成

PEGS4 は通常変更しないので、MORTRAN3 と同様に実行可能ファイルの形で保存する。コンパイルには pgs4compile を用いる (内容は図 7 参照)

```
pgs4compile
```

と入力すると pgs4.mortran をコンパイルし、作成された実行可能ファイルを pgs4.exe として保存する。

コンパイルに失敗した場合には、pgs4.mortran をエディタで開き CALL IOINIT がコメントアウトしてある事を確認する事。その時に、次の OPEN 文が書いてあることも確認する事。

```
OPEN(5,file='pgs4job.pgs4inp');
OPEN(6,file='pgs4job.pgs4lst');
OPEN(7,file='pgs4job.pgs4dat');
OPEN(8,file='pgs4pepr.dat');
OPEN(9,file='pgs4form.dat');
OPEN(10,file='pgs4job.pgs4err');
OPEN(21,file='pgs4job.pgs4plot');
OPEN(22,file='aprime.data');
```

### 3.5.2 PEGS4の実行

pegs4run を用いて peps4.exe を実行し、物質データを作成する。通常は PEGS4 の入力データを用意するが、ここでは、saml4.inp を入力データとして試しに実行する。

```
pegs4run sampl4.inp peps4.output
```

と入力する。ここで、第 1 オペラントが入力ファイル名であり第 2 オペラントが出力ファイル名である。これも同様にファイル名を指定しないと止るので注意。pegs4run と sampl4.inp の内容をそれぞれ図 8 および図 9 に示す。

### 3.5.3 PEGS4の動作確認

PEGS4 は自動的にエネルギー分点を設定し、その分点の間での物理量（全断面積・分岐比など）を直線内挿しその直線の係数を出力する。インストールを行う計算機上での PEGS4 の動作を調べるために基準となる計算機での PEGS4 出力とインストールを行う計算機での PEGS4 出力を較べることが必要であるが、出力を直接較べるのは困難である。計算精度の違いなどによりエネルギー分点が違っていると同一物理量でも見かけ上全く違う表現になるからである。このため examin.mor という EGS4 のユーザーコードを使用する。examin.mor は、PEGS4 データを読み込みそれを表の形で出力し SHOWER を CALL せずに終了するユーザーコードで、

```
pegs4run examin.mor peps4.output
```

と入力すれば実行できる。（PRESTA は使用しない）

ここで作成された mortjob.output6 と examin.out を

```
compare mortjob.output6 examin.out
```

と入力して比較し、両者が全く同一であるか、5 桁目が何箇所か違う位であれば、インストールをおこなった計算機上で PEGS4 が、標準となる計算機上での PEGS4 と同じ動作をしたと考えられる。明かに異なれば原因をさがす必要がある。saml4.inp を入力とし Stom と Israel が編集した光電効果断面積 (pgs4pepr.dat<sup>11</sup>) を用いて、PEGS4 を KEK の共通計算機 (ccce5=日立製 WS) で走らせて、物質データファイルを作成し、その物質データファイルを examin.mor によって分析した結果が examin.out である。その内容を章末 (図 10 および 図 11) に示す。

---

<sup>11</sup> pgs4pepr.dat については、「Unix 計算機へのインストール」の部分の脚注 2 を参照。

## 4 練習問題

### 4.1 実習時の注意事項

- Emacs を使用する場合には、使用前に次のように入力して下さい。

```
% setenv DISPLAY <ip address>:0.0
```

- PEGS4 実行時の IEEE エラーは無視して下さい。
- PC で実習する場合、指定したディレクトリ以外のファイルを操作しないで下さい。また Windows の設定もなるべく変えないで使用して下さい。

### 4.2 EGS4RUN で生成するファイル

EGS4RUN を実行すると、下記のファイルが作成される。

PC 版ファイル名	Unix 版,Mac 版 拡張子	内容
MORTJOB.MOR	.mortran	ユーザーコード + EGS4 (MORTRAN 言語)
MORTJOB.XSE	.xsec	物質データファイルのコピー
MORTJOB.MLT	.list	Mortran ファイルの listing Mortran → Fortran 変換時のエラー情報
MORTJOB.FOR	.f	Fortran ソース
MORTJOB.OUT	.output(6)	計算結果
MORTJOB.DUM	.dummy	HATCH による物質データのエコーバック
MORTJOB.LST		Fortran ファイルの listing Fortran コンパイル時のエラー情報
MORTJOB.OBJ		オブジェクト・ファイル
MORTJOB.MAP		リンカのマップ・ファイル
MORTJOB.EXE	(a.out)	実行ファイル

コンパイル失敗時には、次のファイルを調べる。

- MORTJOB.MOR : ユーザーコードが含まれているか ?
- MORTJOB.MLT : Mortran エラーが発生していないか ?
- MORTJOB.FOR : FORTRAN ソースにマクロが残っていないか ?
- MORTJOB.LST : Fortran コンパイル時エラーが起きていないか ?

Unix の場合、Fortran エラーは画面表示のみでファイルには残らないので、Fortran コンパイル時にエラーを見つけたら、即座に「cntl」キーと「z」キーを同時に押してジョブを停止させ、エラーの行番号と内容をメモする。(mortran や egs4run の f77 や g77 の行に「>& file name」を付け加えてエラーメッセージをファイルに書き込んでもよい。)

### 4.3 MORTRAN の練習問題

1. check77.mor から出発し、50 以下の素数を表示する Mortran プログラムを作成せよ。

(ヒント)

- DO ループの中を「[」と「]」で囲って指定する Mortran 文法を利用せよ。
- $N = \text{MOD}(I, J)$ ; とすれば I を J で割った余りが得られる。  $J < I$  であるすべての J に対して  $N \neq 0$  ならば I は素数である。
- タブを使用すると MORTRAN のエラーとなるので、タブは使用しない事。

2.  $\pi$  の計算

- (a) 2 個の乱数を発生させ、  $-1 < x < 1$ ,  $-1 < y < 1$  の正方形内に均一に点を分布させよ。

(ヒント)

- 次の乱数発生マクロを ucsampl4.mor からコピーして用いよ。

```
REPLACE {$RANDOMSET#;} WITH  
{IXX=IXX*663608941;{P1}=IXX*0.23283064E-09;  
IF (IXX.LT.0){P1}={P1}+1.0;}
```

なお、EGS4 実行時には自動的にこのマクロと同等のマクロを nrcc4mac.mor から読み込むので、ユーザーコード中にこのマクロを書いておく必要はない。

- 次の乱数初期値を ucsampl4.mor からコピーして用いよ。

```
IXX=987654321; "RANDOM NUMBER GENERATOR SEED"
```

- \$RANDOMSET (変数名); で (変数名) に 0-1 の乱数が代入される。

- (b) 上の点の内、  $x^2 + y^2 = 1$  の円の内側にあるものの割合を求め、これから  $\pi$  を求めよ。

3. 点等方線源

- (a) 3 個の乱数を発生させ、  $-1 < x < 1$ ,  $-1 < y < 1$ ,  $-1 < z < 1$  の立法体内に均一に点を分布させよ。

- (b) 上の点の内、  $x^2 + y^2 + z^2 = 1$  の球の内側にあるものに注目すると、原点から等方に分布している。この球の内側の点の座標を、原点から点までの距離で割れば、点等方線源の方向余弦が得られる。この点等方線源の方向余弦を 10 組求めよ。

### 4.4 EGS4, PEGS4 の練習問題

1. NaI 検出器の計算 ucnai3p.mor を次のように変更し、ピーク効率および全効率の変化を調べよ。

- (a) 有感領域の厚さを 2 倍する。

- (b) 線源エネルギーを 667 keV ( $^{137}\text{Cs}$ ) に変更する。(有感領域厚さは元に戻す。)

- (c) 現状では1種類(1.33 MeV)として扱っている $^{60}\text{Co}$ からのガンマ線を、2種類のガンマ線(1.173 MeVと1.333 MeV。強度比は1:1)に変更せよ。  
(ヒント) CALL SHOWER ループ内で乱数を発生させ、その乱数によってどちらかのエネルギーを選択する。
- (d) 現状では一方向(Z方向)のみに放出している線源光子を、点等方線源に変更せよ。  
(ヒント) CALL SHOWER ループ内に Morfran 練習問題で作成した点等方線源を組み込む。

## 2. Ge 検出器の計算

- (a) PEGS4 を用いて Ge の物質データを作成せよ。エネルギー範囲は、 $\gamma$  は 10 keV–50 MeV、 $e^-$  は 521 keV–50 MeV とする。  
(ヒント) sampl4.inp をコピーし、必要事項のみ変更する。
- (b) ucna13p.mor の有感媒質を NaI から Ge に変更せよ。
- (c) (a) と (b) の結果を用いて Ge 検出器の応答関数を求めよ。同じ大きさの NaI 検出器と比べてピーク効率および全効率はどうか異なるか?  
(ヒント) nai.dat 中の Al のデータが必要であるから、コマンドまたはエディタで、Ge のデータの後に Al のデータまたは nai.dat 全体を付け足す事。(例えば、unix では `cat ge.dat nai.dat > ge-al.dat` とする。)

## 3. 空気電離箱の計算

- (a) 空気(20°、1気圧)の物質データを作成せよ。IAPRIM=1を使用せよ。エネルギー範囲は2.(a)と同じとする。
- (b) ucna13p.mor の有感部分を空気に変更せよ。
- (c) 上で作成した空気の物質データを用いて空気電離箱の計算を行い、入射エネルギーのうち空気中でのエネルギー損失となるものの割合を求めよ。(Energy Deposition Summary の出力を使用する。)
- (d) 空気の W 値(33.8 eV/pair)を用いて、1.33 MeV の入射光子1個当たりのこの電離箱の出力(Coulomb/source)を求めよ。  
(ヒント) 空気有感層中でのエネルギー損失を W 値で割り、電荷素量をかければ電離箱の出力(クーロン/入射光子)が得られる。

## 4.5 MORTRAN の練習問題の解答

### 1. 50 以下の素数を表示する Mortran プログラム

```
!COMMENTS;!INDENT C5;!INDENT M4;!INDENT F2;!LIST;
DO I=1,50 [
  IPRIME=1;
  DO J=2,I-1 [
    K=MOD(I,J);
    IF(K.EQ.0)IPRIME=0; "Not a prime number"
  ]
  IF(IPRIME.EQ.1) OUTPUT I; (I3,' is a prime number.');
```

```
STOP;
END;
%%
```

### 2. $\pi$ の計算 (a) と (b) を合わせたプログラムを示す。

```
!COMMENTS;!INDENT C5;!INDENT M4;!INDENT F2;!LIST;
REPLACE {;COMIN/RANDOM/;} WITH {;COMMON/RANDOM/IXX;}
REPLACE {$RANDOMSET#;} WITH
{IXX=IXX*663608941;{P1}=IXX*0.23283064E-09;IF(IXX.LT.0){P1}={P1}+1.;;}
;COMIN/RANDOM/;
IXX=123456789;
INSIDE=0;
NCASES=100000;
DO I=1,NCASES [
  $RANDOMSET X1; X2=X1*2.0-1.0;
  $RANDOMSET Y1; Y2=Y1*2.0-1.0;
  IF(X2*X2+Y2*Y2.LE.1.0)[
    INSIDE=INSIDE+1;
  ]
]
PIVAL=4.0*FLOAT(INSIDE)/FLOAT(NCASES);
OUTPUT PIVAL; (' Estimated pi value='1PE12.5);
STOP;
END;
%%
```

### 3. 点等方線源 (a) と (b) を合わせたプログラムを示す。

```
!COMMENTS;!INDENT C5;!INDENT M4;!INDENT F2;!LIST;
REPLACE {;COMIN/RANDOM/;} WITH {;COMMON/RANDOM/IXX;}
REPLACE {$RANDOMSET#;} WITH
{IXX=IXX*663608941;{P1}=IXX*0.23283064E-09;IF(IXX.LT.0){P1}={P1}+1.;;}

;COMIN/RANDOM/;
IXX=123456789;
NCASES=10;
DO I=1,NCASES [
  :Sampling_start: "Label for goto statement"
  $RANDOMSET X1; X2=X1*2.0-1.0;
  $RANDOMSET Y1; Y2=Y1*2.0-1.0;
```

```

$RANDOMSET Z1; Z2=Z1*2.0-1.0;
R2=SQRT(X2*X2+Y2*Y2+Z2*Z2);
IF(R2.GT.1.0) GOTO :Sampling_start;;
UI=X2/R2; VI=Y2/R2; WI=Z2/R2;
OUTPUT UI,VI,WI; (' UI,VI,WI=',3E11.4);
]
STOP;
END;
%%

```

#### 4.6 EGS4, PEGS4 の練習問題の解答

##### 1. ucnaip3p.mor の変更

- (a) 変更前には、ピーク効率=35.824%、全効率=72.392%であり、厚さを2倍にすると、ピーク効率=52.039%、全効率=92.137%に増加する。変更する部分は、

```
TDE=15.24;
```

- (b) ピーク効率=58.294%、全効率=85.176%となる。変更する部分は、

```
EI=0.667;
```

- (c) ピーク効率=36.647%、全効率=74.255%となる。CALL SHOWERのループの最初の部分を次のように変更する。

```

DO I=1,NCASPB ["START OF SHOWER CALL LOOP OF EACH BATCH"
$RANDOMSET AA;
IF(AA.LT.0.5) [EI=1.33 +ABS(IQI)*PRM;] "TOTAL ENERGY OF PARTICLE"
ELSE [EI=1.17 +ABS(IQI)*PRM;]

```

なお、この例のように複数のエネルギーの入射粒子に対してはECNSVは正しく動作していない。(1.0となるべきエネルギー損失の総計が0.9397となっている。)これを正しく動作させるためには次のように、全運動エネルギーの集計を変更する必要がある。

```
" STEP 6"
TOTKE=0.0;
```

```

"STEP 7"
"After EI is determined"
AVAILP=EI + IQI*PRM; "AVAILABLE K.E. (MeV) (MUST BE REAL*8)"

```

```

"After shower call"
TOTKE=TOTKE+AVAILP; "TOTAL (AVAILABLE) K.E."

```

- (d) ピーク効率=7.2549%、全効率=21.510%となる。有感部分の立体角が $4\pi$ よりも小さい事の影響が現れる。CALL SHOWERのループの最初の部分を次のように変更する。

```

DO I=1,NCASPB ["START OF SHOWER CALL LOOP OF EACH BATCH"
"Point isotropic source"
:Sampling_start: "Label for goto statement"

```

```

$RANDOMSET X1; X2=X1*2.0-1.0;
$RANDOMSET Y1; Y2=Y1*2.0-1.0;
$RANDOMSET Z1; Z2=Z1*2.0-1.0;
R2=SQRT(X2*X2+Y2*Y2+Z2*Z2);
IF(R2.GT.1.0) GOTO :Sampling_start;;
UI=X2/R2; VI=Y2/R2; WI=Z2/R2;

```

## 2. Ge 検出器の計算

- (a) 下の入力データを用いて PEGS4 を実行する。

```

ELEM
  &INP IRAYL=1 &END
GE
GE
ENER
  &INP AE=0.521,UE=50.0,AP=.01,UP=50.0 &END
TEST
  &INP &END
PWL
  &INP &END
DECK
  &INP &END

```

- (b) ユーザーコード中の

```
DATA MEDARR/$S'NAI-IAPRIM',14*' ',
```

を

```
DATA MEDARR/$S'GE',22*' ',
```

と変更せよ。また、

```
IEDGFL(8)=53; "53:Atomic number of I"
```

を

```
IEDGFL(8)=32; "32:Atomic number of Ge"
```

と変更せよ。物質名の後の空白(22\*' 'や14\*' ')は物質名の長さに合わせて24文字になるよう調整する。これを行わないと2個目の物質名が配列の先頭からずれてしまう。

- (c) ピーク効率=37.529%、全効率=85.137%となる。(有感部分の密度が大きいため全効率は増加する。Geの原子番号が、NaIの実効原子番号より小さいため密度増加の影響を打ち消し、ピーク効率はあまり変化しない。)

## 3. 空気電離箱の計算

- (a) 下の入力データを用いて PEGS4 を実行する。

```

MIXT
  &INP NE=3,RHO=1.2929E-3,GASP=0.93174,
  RHOZ=0.75575, 0.23143, 0.01282,

```

```

IAPRIM=1 &END
AIR                                AIR-GAS
N O AR
ENER
  &INP AE=0.521,UE=50.0,AP=.01,UP=50.0 &END
TEST
  &INP &END
PWLF
  &INP &END
DECK
  &INP &END

```

(b) ユーザーコード中の2ヶ所を次のように変更する。

```

DATA MEDARR/$S'AIR',21*' ',
IEDGFL(8)=18; "18:Atomic number of Ar"

```

(c) 領域8での全粒子のエネルギー損失が求める量であり、 $1.268 \times 10^{-4}$  である。

$$(d) 1.33 \times 10^6 (\text{eV/source}) \cdot 1.268 \times 10^{-4} \cdot \frac{1}{33.8 (\text{eV/pair})} \cdot 1.6 \times 10^{-19} (\text{Coulomb/pair}) \\ = 7.98 \times 10^{-19} (\text{Coulomb/source})$$

- 空気有感層でのエネルギー付与の大部分はアルミ製ふたで発生した電子によるものである。実際の電離箱では、実効原子番号が空気と近い物質でふたを作成するか、ふたを使用しない事に注意。
- どの領域でのどの反応で光子から電子に変換されたかを記録しておき、領域別や反応別にスコアする事がCOMMON/STACK中のLATCHという変数を利用することで可能である。LATCHについてはSLAC-265のTutor5参照。
- 上記のIEDGFL(8)=18; ではすべての光電効果を、空気中で最もZの大きい元素であるArによる光電効果であると仮定している。EGS4では特別のプログラムを用いない限り、化合物および混合物中の複数の元素のX線を計算できないため、このような近似を用いている。この問題については、A. Del. Guerra, NIM A306 (1991) 378-385 参照。
- この計算は統計誤差が大きいので、計算機の機種により10-20%の計算結果の違いを生じることがある。

(file: egsins7.tex, egsins7a.tex and egsins7b.tex)

Figure 1: List of egs4run (unix)

---

```
# * E G S 4 R U N *
# Usage: egs4run $1 $2
#       $1: User code name.  ex. ucsampl4.mor
#       $2: Material data file.  ex. sampl4.dat

echo "egs4run for nrcc version started"
echo "user/egs4 configuration entered"; echo " "
rm mortjob.*
rm a.out
echo "Do you want to include MACRO in the mortran list ?
(yes or no. null means yes.)"
read key
case $key in
  no ) cp egs4mac.mortran      mortjob.mortran
        echo "MACRO is not included in mortran list";;
  * ) cp listing.on          mortjob.mortran
        cat egs4mac.mortran  >> mortjob.mortran
        echo "MACRO is included in mortran list";;
esac
echo "Do you use PRESTA ? (yes or no. null means yes.)"
read key
case $key in
  no ) cat nrcc4mac.mortran >> mortjob.mortran
        echo "Default EGS4 is selected";;
  * ) cat nrc4macp.mortran >> mortjob.mortran
        cat presta.mortran  >> mortjob.mortran
        echo "PRESTA is selected";;
esac
cat pairmac.mortran >> mortjob.mortran
cat sunmacs.mortran >> mortjob.mortran
cat kek4mac.mortran >> mortjob.mortran
if test $1
  then cat $1                >> mortjob.mortran  #USER CODE is joined.
  else echo "Key in user code"
        read ucode
        cat $ucode          >> mortjob.mortran
fi
cat kek4.mortran >> mortjob.mortran
cat egs4blok.mortran >> mortjob.mortran
cat egs4.mortran >> mortjob.mortran
if test $2
  then cp $2                mortjob.xsec      #Xsec data is copied
  else echo "Key in data-file"
        read dfile
        cp $dfile          mortjob.xsec      #Xsec data is copied.
fi

echo "mortran3 procedure entered"; echo " "
mortran3
echo "f77 procedure entered"; echo " "
f77 mortjob.f
echo "a.out procedure entered"; echo " "
time a.out &
echo "egs4run procedure finished"
```

---

Figure 2: List of egs4run (PC)

---

```
REM Usage: EGS4RUN %1 %2
REM %1: User Code (ex: ucsampl4.mor)
REM %2: Material Data (ex:sampl4.dat)

DEL MORTJOB.*

REM Produce MORTJOB.MOR.
COPY LISTING.ON+EGS4MAC.MOR+NRCC4MAC.MOR+PAIRMAC.MOR+KEK4MAC.MOR+%1+KEK4.MOR+
EGS4BLOK.MOR+EGS4.MOR MORTJOB.MOR

REM Copy material data.
COPY %2 MORTJOB.XSE

REM Convert from MORTJOB.MOR to MORTJOB.FOR.
MORTRAN3

REM Compile, link and execute MORTJOB.FOR.
REM Uncomment out next line for Lahey Fortran 90
!f90 MORTJOB.FOR -nw -lst
REM Microsoft FORTRAN POWERSTATION
REM FL32 /Ox /WO /G4 MORTJOB.FOR

MORTJOB
```

---

Figure 3: List of egs4run (Macintosh)

---

```
echo "egs4run has started"
delete mortjob.≈|| echo "There aren't any mortjob.≈"

catenate listing.on > mortjob.mortran
catenate egs4mac.mortran >>mortjob.mortran
catenate nrcc4mac.mortran >>mortjob.mortran
catenate pairmac.mortran >>mortjob.mortran
catenate kek4mac.mortran >>mortjob.mortran
catenate {1} >>mortjob.mortran
catenate kek4.mortran >> mortjob.mortran
catenate egs4blok.mortran >> mortjob.mortran
catenate egs4.mortran >>mortjob.mortran

catenate {2} >mortjob.xsec

mortran3
alert 'Wait until mortran3 finishes'
RunPPC mortjob.f -saveall -opt=3
```

---

Figure 4: Output of ucsamp4.mor (Unix, PC and Macintosh)

---

```

BEFORE HATCH CALL ITO=      0
EGS SUCCESSFULLY 'HATCHED' FOR      2 MEDIA.
AFTER HATCH CALL IT1=      0
Elapsed Time (sec)= 0.00000E+00
1
SHOWER RESULTS:

      E          Z          W          IQ   IR   IARG
1000.000      0.000000      1.000000      -1    2   -1
163.9947      3.000000      0.9999890      0    3    3
504.8928      3.000000      0.9987407      0    3    3
1000.000      0.000000      1.000000      -1    2   -1
155.8518      3.000000      0.9998820      0    3    3
1000.000      0.000000      1.000000      -1    2   -1
140.0282      3.000000      0.9987389      0    3    3
126.1480      3.000000      0.9998272      0    3    3
202.7643      3.000000      0.9996023      0    3    3
1000.000      0.000000      1.000000      -1    2   -1
116.1481      3.000000      0.9994835      0    3    3
139.3819      3.000000      0.9988134      0    3    3
1000.000      0.000000      1.000000      -1    2   -1
107.5314      3.000000      0.9994128      0    3    3
1000.000      0.000000      1.000000      -1    2   -1

INCIDENT TOTAL ENERGY OF ELECTRON=      1000.0 MEV
IRON SLAB THICKNESS= 3.000 CM
NUMBER OF CASES IN RUN=      10
LAST RANDOM NUMBER= -687420815

ENERGY DEPOSITION SUMMARY:

FRACTION IN REGION  1= 0.0000000
FRACTION IN REGION  2= 0.3735966
FRACTION IN REGION  3= 0.6264034

TOTAL ENERGY FRACTION IN RUN=      1.000000
WHICH SHOULD BE CLOSE TO UNITY
AFTER CALCULATION FINISHED IT2=      0
Elapsed Time (sec)= 0.00000E+00

```

---

This output is a result of calculation using egs4.mor of Ver-2002-7-30-1600 KEK or after. If you use older version of egs4.mor, LAST RANDOM NUMBER=1903435093 and several other points are different.

Figure 5: List of pegs4compile (Unix)

---

```
rm mortjob.*
cp pegs4.mortran mortjob.mortran
echo "mortran3 procedure entered"; echo""
mortran3
echo "f77 procedure entered"; echo""
f77 mortjob.f
mv a.out pegs4.exe
echo "pegs4compile procedure finished"
```

---

Figure 6: List of PEGS4COM.BAT (PC)

---

```
del mortjob.*
copy pegs4.mor mortjob.mor
echo "mortran3 procedure entered"; echo " "
mortran3

echo "Fortran compile procedure entered"; echo " "
REM following two lines are for Lahey F77L-EM/32
f7713 MORTJOB.FOR /H
386link mortjob -symbol
REM following is for Lahey fortran90
REM lf90 MORTJOB.FOR -nw -lst
REM following is for Microsoft Fortran Power station
REM FL32 /WO MORTJOB.FOR

ren mortjob.exe pegs4.exe
echo "pegs4.compile procedure finished"
```

---

Figure 7: List of pegs4compile (Macintosh)

---

```
echo "pegs4compile has started"
delete mortjob.≈|| echo "There aren't any mortjob.≈"
catenate pegs4.mortran > mortjob.mortran

echo "mortran3 procedure entered"; echo " "
mortran3
alert 'Wait until mortran3 finishes'
echo "fortran.ppc procedure entered"; echo " "
fortran.ppc mortjob.f -saveall -opt=3

linkfortranppc pegs4.exe mortjob.f.o
echo "pegs4.compile procedure has finished"
```

---

Figure 8: List of pegs4run (Macintosh)

---

```
echo "pegs4run procedure has started"; echo " "
delete pgs4job.≈|| echo "There aren't any pgs4job.≈"

catenate {1} > pgs4job.pegs4inp

echo "pegs4.exe has started"
pegs4.exe
alert 'Wait until pegs4.exe finishes'
echo "pegs4.exe has finished."
catenate pgs4job.pegs4dat > {2}

echo "pgs4run procedure has finished"
```

---

Figure 9: List of sampl4.inp

Unix, PC(Lahey) and Macintosh

---

```
ELEM
&INP IRAYL=1 &END
FE                               FE
FE
ENER
&INP AE=1.5,UE=20000.,AP=0.1,UP=20000. &END
TEST
&INP &END
PWLF
&INP &END
DECK
&INP &END
```

---

PC(MS-Fortran/Power station)

---

```
ELEM
&INP IRAYL=1 /
FE                               FE
FE
ENER
&INP AE=1.5,UE=20000.,AP=0.1,UP=20000. /
TEST
&INP /
PWLF
&INP /
DECK
&INP /
```

---

OCALL HATCH  
 RAYLEIGH DATA AVAILABLE FOR MEDIUM 1 BUT OPTION NOT REQUESTED.

EGS SUCCESSFULLY 'HATCHED' FOR ONE MEDIUM.  
 CALL TO HATCH COMPLETED

1 SUMMARY OF PHOTON DATA STORED FOR MEDIUM=FE-RAYLEIG  
 ELECTRON KINETIC ENERGY RANGE: 0.989 19999.488 MEV PHOTON ENERGY RANGE: 0.100 20000.000 MEV  
 RADIATION LENGTH= 1.76084 CM DENSITY= 7.8600 G/CM\*\*3

PHOTON ENERGY	GAMMA MEAN FREE PATH(cm)	PHOTOELECTRIC	COMPTON	PAIR	TOTAL	
					cm**2/G	cm**-1
0.100	0.3744	0.59346	0.40654	0.0	3.398E-01	2.671E+00
0.125	0.5482	0.43702	0.56298	0.0	2.321E-01	1.824E+00
0.150	0.6984	0.31741	0.68259	0.0	1.822E-01	1.432E+00
0.200	0.9215	0.17492	0.82508	0.0	1.381E-01	1.085E+00
0.300	1.1971	0.06787	0.93213	0.0	1.063E-01	8.353E-01
0.400	1.3837	0.03478	0.96522	0.0	9.195E-02	7.227E-01
0.500	1.5365	0.02105	0.97895	0.0	8.280E-02	6.508E-01
0.600	1.6724	0.01456	0.98544	0.0	7.608E-02	5.980E-01
0.700	1.7975	0.01086	0.98914	0.0	7.078E-02	5.563E-01
0.800	1.9158	0.00852	0.99148	0.0	6.641E-02	5.220E-01
1.000	2.1367	0.00587	0.99413	0.0	5.954E-02	4.680E-01
1.250	2.3847	0.00426	0.99222	0.0035	5.335E-02	4.193E-01
1.500	2.6152	0.00332	0.98847	0.0082	4.865E-02	3.824E-01
2.000	2.9910	0.00234	0.96428	0.0334	4.254E-02	3.343E-01
3.000	3.5194	0.00152	0.89238	0.1061	3.615E-02	2.841E-01
4.000	3.8473	0.00113	0.81360	0.1853	3.307E-02	2.599E-01
5.000	4.0505	0.00090	0.73966	0.2594	3.141E-02	2.469E-01
6.000	4.1703	0.00074	0.67312	0.3261	3.051E-02	2.398E-01
7.000	4.2389	0.00062	0.61473	0.3846	3.001E-02	2.359E-01
8.000	4.2737	0.00053	0.56420	0.4353	2.977E-02	2.340E-01
10.000	4.2776	0.00017	0.48090	0.5189	2.974E-02	2.338E-01
12.500	4.2181	0.00000	0.40219	0.5978	3.016E-02	2.371E-01
15.000	4.1374	0.00000	0.34410	0.6559	3.075E-02	2.417E-01
20.000	3.9642	0.0	0.26451	0.7355	3.209E-02	2.523E-01
30.000	3.6847	0.0	0.17882	0.8212	3.453E-02	2.714E-01
40.000	3.4782	0.0	0.13400	0.8660	3.658E-02	2.875E-01
50.000	3.3234	0.0	0.10681	0.8932	3.828E-02	3.009E-01
60.000	3.2029	0.0	0.08862	0.9114	3.972E-02	3.122E-01
70.000	3.1130	0.0	0.07577	0.9242	4.087E-02	3.212E-01
80.000	3.0401	0.0	0.06625	0.9337	4.185E-02	3.289E-01
100.000	2.9321	0.0	0.05303	0.9470	4.339E-02	3.411E-01
125.000	2.8384	0.0	0.04253	0.9575	4.482E-02	3.523E-01
150.000	2.7695	0.0	0.03555	0.9644	4.594E-02	3.611E-01
200.000	2.6754	0.0	0.02686	0.9731	4.755E-02	3.738E-01
300.000	2.5701	0.0	0.01819	0.9818	4.950E-02	3.891E-01
400.000	2.5117	0.0	0.01384	0.9862	5.065E-02	3.981E-01
500.000	2.4742	0.0	0.01123	0.9888	5.142E-02	4.042E-01
600.000	2.4478	0.00000	0.00946	0.9905	5.198E-02	4.085E-01
700.000	2.4282	0.0	0.00820	0.9918	5.240E-02	4.118E-01
800.000	2.4129	0.00000	0.00723	0.9928	5.273E-02	4.144E-01
1000.000	2.3906	0.0	0.00589	0.9941	5.322E-02	4.183E-01
1250.000	2.3719	0.00000	0.00480	0.9952	5.364E-02	4.216E-01
1500.000	2.3588	0.00000	0.00406	0.9959	5.394E-02	4.239E-01
2000.000	2.3415	0.0	0.00311	0.9969	5.434E-02	4.271E-01
3000.000	2.3231	0.0	0.00215	0.9979	5.477E-02	4.305E-01
4000.000	2.3131	0.0	0.00165	0.9983	5.500E-02	4.323E-01
5000.000	2.3069	0.0	0.00135	0.9987	5.515E-02	4.335E-01
6000.000	2.3026	0.0	0.00114	0.9989	5.525E-02	4.343E-01
7000.000	2.2994	0.0	0.00099	0.9990	5.533E-02	4.349E-01
8000.000	2.2970	0.0	0.00088	0.9991	5.539E-02	4.354E-01
10000.000	2.2934	0.0	0.00071	0.9993	5.548E-02	4.360E-01
12500.000	2.2905	0.00000	0.00058	0.9994	5.555E-02	4.366E-01
15000.000	2.2884	-0.00000	0.00050	0.9995	5.560E-02	4.370E-01
20000.000	2.2858	0.0	0.00038	0.9996	5.566E-02	4.375E-01

Figure 10: Output of examin.mortran (Photon part. Unix, PC and Macintosh).  
 Line-width is shortened for a printing. Use examin.out file for diff command.

1 SUMMARY OF ELECTRON DATA STORED FOR MEDIUM=FE-RAYLEIG  
 OELECTRON KINETIC ENERGY RANGE: 0.989 19999.488 MEV PHOTON ENERGY RANGE: 0.100 20000.000 MEV  
 ORADIATION LENGTH= 1.76084 cm DENSITY= 7.8600 g/cm\*\*3  
 0200.\*TEFFO= 0.0594 cm

OKINETIC ENERGY MeV	STEP SIZE cm	e(-) DEDX MeV/cm	e(+) DEDX MeV/cm	ENERGY LOSS	e(-),e(+) MEAN FREE PATH(BREM FRACTION) cm
0.989	1.001E-02	10.368	10.113	0.105	9.173E-01( 1.00000) 6.797E-01( 0.74102)
1.000	1.018E-02	10.363	10.104	0.105	9.114E-01( 1.00000) 6.761E-01( 0.74190)
1.250	1.426E-02	10.311	9.971	0.118	8.065E-01( 1.00000) 6.129E-01( 0.75989)
1.500	1.882E-02	10.325	9.920	0.130	7.398E-01( 1.00000) 5.724E-01( 0.77376)
2.000	2.941E-02	10.416	9.919	0.153	6.497E-01( 0.99340) 5.193E-01( 0.79372)
3.000	5.614E-02	10.340	10.021	0.193	4.802E-01( 0.86472) 4.542E-01( 0.81792)
4.000	5.940E-02	10.375	10.133	0.154	4.174E-01( 0.84114) 4.129E-01( 0.83189)
5.000	5.940E-02	10.426	10.228	0.124	3.819E-01( 0.83511) 3.845E-01( 0.84068)
6.000	5.940E-02	10.475	10.308	0.104	3.563E-01( 0.83526) 3.615E-01( 0.84758)
7.000	5.940E-02	10.519	10.374	0.089	3.377E-01( 0.83698) 3.440E-01( 0.85261)
8.000	5.940E-02	10.558	10.430	0.078	3.237E-01( 0.83910) 3.304E-01( 0.85634)
10.000	5.940E-02	10.624	10.520	0.063	3.040E-01( 0.84330) 3.105E-01( 0.86151)
12.500	5.940E-02	10.687	10.603	0.051	2.855E-01( 0.84896) 2.915E-01( 0.86665)
15.000	5.940E-02	10.737	10.666	0.043	2.722E-01( 0.85375) 2.775E-01( 0.87044)
20.000	5.940E-02	10.809	10.756	0.032	2.536E-01( 0.86131) 2.579E-01( 0.87590)
30.000	5.940E-02	10.897	10.861	0.022	2.322E-01( 0.87106) 2.353E-01( 0.88249)
40.000	5.940E-02	10.948	10.921	0.016	2.194E-01( 0.87733) 2.218E-01( 0.88672)
50.000	5.940E-02	10.982	10.960	0.013	2.106E-01( 0.88183) 2.125E-01( 0.88979)
60.000	5.940E-02	11.005	10.987	0.011	2.039E-01( 0.88533) 2.055E-01( 0.89225)
70.000	5.940E-02	11.023	11.007	0.009	1.986E-01( 0.88813) 2.000E-01( 0.89427)
80.000	5.940E-02	11.037	11.023	0.008	1.943E-01( 0.89045) 1.955E-01( 0.89596)
100.000	5.940E-02	11.056	11.044	0.007	1.875E-01( 0.89410) 1.884E-01( 0.89870)
125.000	5.940E-02	11.070	11.062	0.005	1.812E-01( 0.89752) 1.820E-01( 0.90134)
150.000	5.940E-02	11.080	11.073	0.004	1.764E-01( 0.90015) 1.770E-01( 0.90343)
200.000	5.940E-02	11.091	11.085	0.003	1.694E-01( 0.90402) 1.699E-01( 0.90657)
300.000	5.940E-02	11.098	11.095	0.002	1.605E-01( 0.90896) 1.609E-01( 0.91075)
400.000	5.940E-02	11.100	11.097	0.002	1.548E-01( 0.91216) 1.551E-01( 0.91355)
500.000	5.940E-02	11.100	11.098	0.001	1.507E-01( 0.91448) 1.509E-01( 0.91561)
600.000	5.940E-02	11.100	11.098	0.001	1.475E-01( 0.91628) 1.476E-01( 0.91725)
700.000	5.940E-02	11.100	11.099	0.001	1.449E-01( 0.91775) 1.450E-01( 0.91858)
800.000	5.940E-02	11.100	11.099	0.001	1.427E-01( 0.91897) 1.428E-01( 0.91971)
1000.000	5.940E-02	11.100	11.099	0.001	1.392E-01( 0.92094) 1.393E-01( 0.92154)
1250.000	5.940E-02	11.100	11.099	0.001	1.359E-01( 0.92281) 1.360E-01( 0.92330)
1500.000	5.940E-02	11.100	11.100	0.000	1.334E-01( 0.92427) 1.334E-01( 0.92469)
2000.000	5.940E-02	11.100	11.100	0.000	1.295E-01( 0.92647) 1.295E-01( 0.92679)
3000.000	5.940E-02	11.100	11.100	0.000	1.244E-01( 0.92935) 1.244E-01( 0.92957)
4000.000	5.940E-02	11.100	11.100	0.000	1.210E-01( 0.93126) 1.211E-01( 0.93142)
5000.000	5.940E-02	11.100	11.100	0.000	1.185E-01( 0.93266) 1.186E-01( 0.93280)
6000.000	5.940E-02	11.100	11.100	0.000	1.166E-01( 0.93378) 1.166E-01( 0.93389)
7000.000	5.940E-02	11.100	11.100	0.000	1.150E-01( 0.93469) 1.150E-01( 0.93478)
8000.000	5.940E-02	11.100	11.100	0.000	1.136E-01( 0.93544) 1.136E-01( 0.93554)
10000.000	5.940E-02	11.100	11.100	0.000	1.114E-01( 0.93669) 1.115E-01( 0.93676)
12500.000	5.940E-02	11.100	11.100	0.000	1.093E-01( 0.93790) 1.093E-01( 0.93794)
15000.000	5.940E-02	11.100	11.100	0.000	1.077E-01( 0.93882) 1.077E-01( 0.93888)
19999.488	5.940E-02	11.100	11.100	0.000	1.052E-01( 0.94026) 1.052E-01( 0.94030)

OWHAT FRACTION SHOULD TMXS BE:

Figure 11: Output of examin.mortran (Electron part. Unix, PC and Macintosh).  
 Line-width is shortened for a printing. Use examin.out file for diff command.